

**Amendments to the Specification:**

Please replace the paragraph beginning at page 11, line 3 to line 8, with the following rewritten paragraph:

Referring again to SINGLE FILE NAME EXTENSION check operation 212, alternatively, upon a determination that there is not a single file name extension ("NO"), e.g., the file name includes multiple file name extensions, SINGLE FILE NAME EXTENSION check operation 212, optionally, transitions processing to a BY-PASS OPTION SELECTED check operation 216.

Please replace the paragraph beginning at page 13, line 11 to line 18, with the following rewritten paragraph:

Referring again to SINGLE FILE NAME EXTENSION check operation 212, when the by-pass option is not utilized, upon a determination that there is not a single file name extension ("NO"), SINGLE FILE NAME EXTENSION check operation 212 transitions processing to DANGEROUS FILE NAME EXTENSION check operation 220 and BY-PASS OPTION SELECTION check operation 216 and FILE NAME EXTENSION VISIBLE check operation 218 are not performed.

Please replace the paragraph beginning at page 13, line 19 to line 30, with the following rewritten paragraph:

FIG. 4 illustrates a process flow diagram of a host computer method 400 used in dangerous file name extension check operation 220 of FIG. 2 in accordance with one embodiment of the present invention. Referring to FIGS. 1, 2, 3 and 4 together,

in one embodiment, as earlier described with reference to method 200 and FIG. 2, method 400 is entered at ENTER operation 402 from SINGLE FILE NAME EXTENSION check operation 212, from optional BY-PASS OPTION SELECTED check operation 216, or from optional FILE NAME EXTENSION VISIBLE check operation 218. From ~~enter~~ ENTER operation 402, processing transitions to a DETERMINE LAST FILE NAME EXTENSION operation 404.

Please replace the paragraph beginning at page 15, line 9 to line 22, with the following rewritten paragraph:

For example, in a Windows®32 operating system, the file is located using the file name (operation 208). The file is opened and the file contents are accessed and examined to determine if a Portable Executable (PE) header is present, such as by parsing the contents of the file. Accessing file contents, content parsing, PE file format and PE headers are well known to those of skill in the art and are not further described herein for clarity of description. When the PE header is present, the file is determined to be executable, and thus, the last file name extension is determined to be an executable file name extension. Alternatively, when the PE header is not present, the file is determined to not be executable, and thus, the last file name extension is determined to not be an executable file name extension.